WePS-3 Evaluation Campaign: Overview of the Web People Search Clustering and Attribute Extraction Tasks

Javier Artiles[1], Andrew Borthwick[2], Julio Gonzalo[1], Satoshi Sekine[3], and Enrique Amigó[1]

[1] NLP Group of UNED University,
Madrid, Spain
http://nlp.uned.es
[2] Intelius, Inc.
Bellevue, WA, USA
http://search.intelius.com
[3] CS Dept., New York University
New York, USA
http://www.nyu.edu

Abstract. The third WePS (Web People Search) Evaluation campaign took place in 2009-2010 and attracted the participation of 13 research groups from Europe, Asia and North America. Given the top web search results for a person name, two tasks were addressed: a clustering task, which consists of grouping together web pages referring to the same person, and an extraction task, which consists of extracting salient attributes for each of the persons sharing the same name. Continuing the path of previous campaigns, this third evaluation aimed at merging both problems into one single task, where the system must return both the documents and the attributes for each of the different people sharing a given name. This is not a trivial step from the point of view of evaluation: a system may correctly extract attribute profiles from different URLs but then incorrectly merge profiles. This campaign also featured a larger testbed and the participation of a state-of-the-art commercial WePS system in the attribute extraction task. This paper presents the definition, resources, evaluation methodology and results for the clustering and attribute extraction tasks.

Keywords: Web Search, Web People Search, Text Clustering, Attribute Extraction, Meta-search Engines, Evaluation

1 Introduction

The Web People Search task has been defined in WePS campaigns as the problem of organizing web search results for a given person name. The most frequently used web search engines return a ranked list of URLs which typically refer to various people sharing the same name. Ideally, the user would obtain groups of documents that refer to the same individual, possibly with a list of person attributes that help the user choosing the cluster that represents the person she is looking for.

From a practical point of view, the people search task is highly relevant: between 11 and 17% of web queries include a person name, 4% of web queries are just a person name, and person names are highly ambiguous: according to the US Census Bureau, only 90,000 different names are shared by more than 100,000,000 people. An indirect proof of the relevance of the problem is the fact that, since 2005, a number of web startups have been created precisely to address it (Spock.com - now Intelius - and Zoominfo.com being the best known).

From a research point of view, the task is challenging (the number of clusters is not known a priori; the degree of ambiguity does not seem to follow a normal distribution; and web pages are noisy sources from which attributes and other indexes are difficult to extract) and has connections with Natural Language Processing and Information Retrieval tasks (Text Clustering, Information Extraction, Word Sense Discrimination) in the context of the WWW as data source.

WePS-1 [3] was run as a Semeval 1 task in 2007, receiving submissions from 16 teams (being one of the largest tasks in Semeval) and WePS-2 [4] was run as a workshop of the WWW 2009 Conference, with the participation of 19 research teams. In the first campaign we addressed only the name co-reference problem, defining the task as clustering of web search results for a given person name. In the second campaign we refined the evaluation metrics [2] [1] and added an attribute extraction task [15] for web documents returned by the search engine for a given person name.

For this third campaign we aimed at merging both problems into one single task, where the system must return both the documents and the attributes for each of the different people sharing a given name. This is not a trivial step from the point of view of evaluation: a system may correctly extract attribute profiles from different URLs but then incorrectly merge profiles [4]

WePS-1 and WePS-2 focused on consolidating a research community around the problem and an optimal evaluation methodology. In WePS-3 the focus was on implicating industrial stakeholders in the evaluation campaign, as providers of input to the task design phase and also as providers of realistic scale datasets. To reach this goal, we have incorporated as co-coordinator Andrew Borthwick, principal scientist at Intelius, Inc. – one of the main Web People Search services – which provides advanced people attribute extraction and profile matching from web pages.

This paper presents an overview of the WePS-3 clustering and attribute extraction tasks. The task definition is provided in Section 2, the WePS-3 testbed is described in Section 3, the methodology to produce our gold-standard is ex-

[4] WePS-3 also included a new task focused on the ambiguity of organization names. Name ambiguity for organizations is a highly relevant problem faced by Online Reputation Management systems. For a full description of this task and the results of the evaluation please refer to [8].

plained in Section 4 and Section 5 includes the evaluation metrics and the campaign design. We also provide an overview of the participating systems and the results of the evaluation in Section 6. Finally, we end with some concluding remarks in Section 7.

2 Task Definition

Given a set of web search results obtained using a person name as query, the proposed tasks are to cluster these search results according to the different people sharing the name and to extract certain biographical attributes for each person (i.e., for each cluster of documents). Groups were allowed to perform only the clustering task, or both tasks together.

Compared to previous WePS campaigns, the clustering task is defined in the same way, but the testbed is larger and more diverse (see Section 3). Also there is a closer relation between the clustering and attribute extraction tasks. The WePS-3 Attribute Extraction task is different from WePS-2 in that systems are requested to relate each attribute to a person (cluster of documents) instead of just listing the attributes obtained from each document. This is the reason why participants in the AE task are required to participate in the Clustering task too. Systems are expected to output one attribute of each type in each cluster of documents (i.e. only one affiliation, only one occupation, etc. for each person).

All attributes listed in Table 1 were included in the attribute extraction task[5].

Attribute class	Example of attribute value
Date of birth	4 February 1888
Birth place	Brookline, Massachusetts
Other name	JFK
Occupation	Politician
Affiliation	University of California, Los Angeles
Award	Pulitzer Prize
School	Stanford University
Major	Mathematics
Degree	Ph.D.
Mentor	Tony Visconti
Nationality	American
Relatives	Jacqueline Bouvier
Phone	+1 (111) 111-1111
FAX	(111) 111-1111
Email	xxx@yyy.com
Web site	http://nlp.cs.nyu.edu

Table 1. Table 1 Definition of 16 attributes of Person at WePS-2

[5] Please refer to the WePS-3 Attribute Extraction Task Guidelines in the WePS website (http://nlp.uned.es/weps) for a detailed definition of each attribute.

Compared to the WePS-2 Attribute Extraction there were two main modifications: (i) WePS-2 training data had an attribute "education", which was separated into three attributes "school", "degree" and "major" in the test data. WePS-3 use school/degree/major as independent attributes; (ii) The annotated data in WePS-2 included "work" and "location", but these were not used in the WePS-2 evaluation and were not considered in WePS-3.

3 Data sets

3.1 Clustering Training Dataset

Participants used the WePS-1 and WePS-2 public clustering testbeds to develop their systems. These datasets consist of the top web search results for a number different ambiguous person names, and contain human assessments of the correct way to group these documents according the different people mentioned with the same name (see [3, 4] for a detailed explanation of the corpus creation and annotation guidelines). The output format remained as in WePS-2 (a "clustering" root element and "entity" elements for each cluster of documents), except for a slight change in the XML format: now "doc" elements associated to a person are enclosed in a "documents" element, as in the Figure 1.

```
<entity id="1">
 <documents>
   <doc rank="99" />
   <doc rank="104" />
 </documents>
</entity>
```

Fig. 1. Sample output from the clustering task.

3.2 Attribute Extraction Training Dataset

A training dataset was provided for the Attribute Extraction Task based on the WePS-2 clustering and attribute extraction test datasets (see [15]). Given the clustering gold standard and attributes extracted for documents in the WePS-2 corpus, we generated a view of the extracted attributes grouped by cluster instead of documents. This provided the participants with the kind of output expected from their systems in WePS-3.

Both the clustering and attribute extraction output were provided in the same XML file (see Figure 2). In this file each cluster of documents is specified by the element entity, which contains the list of grouped documents and the list of extracted attributes. For each attribute it's required to indicate the type of attribute (date_of_birth, occupation, etc.), the source from which it was extracted (document ranking) and the value.

```
<clustering searchString="AMANDA LENTZ">
 <entity id="16">
   <documents>
     <doc rank="17" />
     <doc rank="66"/>
     <doc rank="73"/>
     <doc rank="51" notes= "from Huron" />
   </documents>
   <attributes>
     <attr type="date\_of\_birth" source="17">4th August 1979</attr>
     <attr type="occupation" source="17">Painter</attr>
   </attributes>
 </entity>
 [...]
</clustering>
```

Fig. 2. Sample output from the combined clustering and attribute extraction tasks.

3.3 Test Dataset

In WePS-3 we decided to substantially increase the amount of test data both in number of documents and person names. The same dataset was used both for the clustering and the attribute extraction tasks. A total of 300 person names were used, compared to 30 names used in WePS-2. As we did in WePS-2, we obtained names randomly from the US Census, Wikipedia and computer science conference program committees. In addition to that, we included names for which at least one person has one of the following occupations: attorney, corporate executive or realtor. 50 names were extracted from each one of these sources to make a total of 300 names.

In order to obtain person names where at least one person in the results sets has a particular occupation we designed a simple algorithm. Given a small set of keywords related to the occupation we are interested in (e.g. “real estate“ or “housing” for realtor) we launch a query to a web search engine and randomly show documents to an annotator until she finds one that refers to a person with the intended occupation. Then we formulate a search query with that person's name. If the reference document is present in the top 200 search results we add this name and these documents to our dataset.

For each name the top 200 web search results from the Yahoo! API [6] were downloaded and archived with their corresponding search metadata (search snippet, title, URL and position in the results ranking).

4 Assessments

Systems are requested to make clusters as accurate as possible over the whole set of documents. However, given the annotation load required to manually cluster

[6] http://developer.yahoo.com/search/

this amount of information, the evaluation was performed only on two people per person name. This allowed us to simplify the annotation task from grouping a large set of documents in an unknown number of people clusters to a classification task where only two people are considered when examining each document in the results. Even for this simplified annotation task a large amount of human resources and time is required. To leverage this problem we used the services of Mechanical Turk, distributing the task among many non-expert workers around the world (see Section 4.1).

Before handing the test data to the annotators, we had to select the two people ("person_a" and "person_b") that would be considered for each person name. In each case, we chose a document in the search results as a reference to classify other documents about that particular person. In general "person_a" is related to the source from which the name was selected (e.g. a Wikipedia person when the source is Wikipedia, a realtor when the source is realtor names, etc.), while "person_b" can be any other person in the search results.

To select a "person_a" reference document, we randomly iterate through the search results until one of the following conditions is satisfied: (i) for Wikipedia names we select one of the Wikipedia articles within the search results for that name; (ii) for computer researchers we select a page that mentions the researcher [7] (iii) occupation-related names already have a reference document that we obtained as described in Section 3.3.

"person_b" can be anybody mentioned with the ambiguous name in the results that does not share the distinctive feature of the first person (not a Wikipedia entity or not having the lawyer, executive or realtor occupation). This second person is also selected by randomly iterating through the results until the conditions are satisfied by a certain web page. In the case of the Census names the only requirement for the two selected people is that they be different, but no constraints are set regarding the characteristics of the person. Finally, for researchers names we found that most of them monopolize search results and hence we did not extract a second person for these names. Still, a name disambiguation system has to be able to recognize that most of these documents belong to one individual and so we kept these names in the dataset.

Once we have the reference documents for all the names in the collection we can proceed to the annotation process. Each worker will receive a set of ten search results for a person name and two reference documents (one describing "person_a" and other describing "person_b"). The task for the annotator is to classify each of the ten documents as referring to either one of the two selected people or to a generic "someone else".

In Table 2 we show the average number of pages assigned to each person on each source of name. The main result to highlight in this table is that people in the conference source tend to monopolize the search results, followed by people that appear in Wikipedia articles. Note that the average number of pages is well below the total pages in the test dataset. The reason for this is that this table

[7] Note that computer scientist names were obtained from a list of conference program committee members, so we already know the researcher's identity

only considers pages for which at least three annotators out of five agreed in the assignment.

name	Avg. number of pages classified as:			
source	person_a	person_b	someone else	total
attorney	5.34	5.66	44.64	55.64
realtor	6.36	4.96	119.56	130.88
executive	7.48	4.20	58.12	69.80
census	4.58	3.00	19.64	27.22
conference	**28.94**	-	26.12	55.06
wikipedia	**9.32**	2.82	23.38	35.52

Table 2. Average number classified of pages by source of the name

For the evaluation of the Attribute Extraction task we didn't rely on a previously generated gold standard. Instead we pooled the output of the participating systems and submitted this for annotation in Mechanical Turk. We only evaluated the extraction of attributes for the same people annotated in the clustering task. For this reason, we only added to the pool attributes whose source is one of the documents annotated as mentioning “person_a“ or “person_b“ according to the Clustering gold standard. The annotators were given one webpage and a set of up to ten $< attribute, value >$ pairs and were asked to decide whether each attribute fell into one of the following categories:

- **Correct** (this is a correct attribute that describes the person in the page).
- **Incorrect** for any reason other than being too long or too short. For instance: The type of attribute is incorrect (e.g. gardener is incorrectly identified as a date of birth); the attribute is not attached to this person (e.g. this attribute describes some other person describe on the page); or the attribute was simply not found in the text describing the subject person.
- Correct, but too long or too short. The attribute is correct, but has one of the following problems:
  - **Too short**. The attribute is incomplete (e.g. ”director” when it should say ”director of marketing”).
  - **Too long**. The attribute contains a correct value but includes irrelevant information (e.g. ”CEO in 1982” when it should say ”CEO”).
- Impossible to tell because **the web page is unreadable**.
- The web page is readable, but **the specified person is not on this page**.

4.1 Mechanical Turk Methodology

For the annotation of the test data we used Mechanical Turk [8]. Mechanical Turk (MTurk) offers a web portal to post tasks known as HITs (Human Intelligence

[8] https://www.mturk.com/mturk/welcome

Tasks). Workers (known as "Turkers") can choose among the available tasks and complete them for a small fee for each task. The use of MTurk for NLP tasks has been studied before [16, 6] and has been found to be effective, but this evaluation forced us to focus on the problem of detecting "spam" annotations by Turkers and to focus on "employee relations" issues of how best to motivate and interact with Turkers.

Spam annotations occur when a Turker attempts to rapidly do a large number of HITs without making a serious attempt at doing quality annotations or by simply writing an automated script to do the HITs. In the following paragraphs, we describe efforts we undertook to discourage spam annotations and to encourage the highest quality workers.

The primary guard that MTurk has to encourage high-quality annotations is the Turker's "HIT approval rate" (HAR), which is the percentage of each Turker's HITs that have been approved divided by the number of approved + rejected HITs. Amazon's instructions on the web site recommend that HITs be posted requiring that Turker's HITs have a minimum HAR of 95%. For our initial annotations, we adopted this recommendation. This succeeded initially, but we found some batches that showed obvious signs of spamming. Table 3 shows average inter-annotator agreement obtained in different portions of the corpus according to the source of the name. For each of the annotated web pages, agreement is measured as the percentage of annotators (five in each case) that selected the most voted annotation. The table also shows the number of HITs generated in each case (each HIT contains 10 pages related to an ambiguous name) and the average number of seconds spent by the annotators working on each HIT. Specifically, as can be seen in Table 3, the census batch had 14/4325 HITs done by workers with average inter-annotator agreement of 50% or higher, while the realtor batch, which had the highest rate of IA agreement and which took the longest time per HIT, had 1,386/4,695 HITs done by workers with IAA of 50% or higher.

source	avg.agreement	avg.time per hit	#hits
attorney	0.51	158.16	4120
census	0.48	54.46	4195
conference	0.50	76.87	4265
executives	0.55	163.04	4240
realtor	0.68	314.45	4695
wikipedia	0.48	131.33	4325

Table 3. Annotation statistics for the clustering task

Concerned that the some of our batches might have suffered from spam, we instituted a number of changes in our MTurk methodology for the attribute annotation task.

1. We raised the minimum HAR to 97%.

2. We added an additional requirement that the Turker must have had at least 500 approved HITs before doing our HITs.
3. On each HIT, we added at least one attribute which we knew to be very likely false (it was an attribute drawn from a response for a different person. Hence it would only be true in the very unlikely case that the two people had the same attribute by chance).
4. We monitored the accuracy of the Turkers on each batch of HITs. Turkers who marked too many of the ”trick attributes” as correct had their work ”rejected” (not paid for and resubmitted to other Turkers to be redone) and were ”blocked”. Blocked workers are barred from ever performing work on the account that blocked them.
5. In addition, we instituted a bonus program whereby we paid cash bonuses to those workers who got the best score on the ”trick” attributes. The workers with the best score received a 100% bonus. At the discretion of the manager of the Turk project (Dr. Borthwick), Turkers with a score close to the best were sometimes given 50% bonuses.
6. Finally, we established a dialog with the Turkers as described below.

We established a dialog with the Turkers in two ways. First of all, we took care to notify the Turkers that we were monitoring them and would reward good workers with bonuses and punish evil-doers with rejection of their HITs and by blocking them from doing future work. We put this notice on every HIT and we also alluded to it on the tag line of the HIT, where we put ”Bonus available!” after the HIT description. Note that the rejection of an entire batch of HITs can have severe consequences for a Turker as it can push his/her HAR below the 95% threshold required to get work.

The more interesting initiative, though, was the establishment of a two-way dialog with the Turkers by posting on the Turker Nation bulletin board [9]. We understand Turker Nation to be the most popular venue for this kind of discussion [10]. As per the convention on Turker Nation, we used a single thread as a point of discussion for all of the attribute extraction HITs. We posted notices there every time a new batch of HITs was posted to Mechanical Turk and we listed the ID's of those workers who received bonuses so as to communicate to Turkers that we were following through on our bonus commitment. Furthermore, we used this board to field queries about how best to judge the HITs. In all, we made 39 posts to this thread and fielded 47 questions and comments from Turkers. Anecdotally, we believe that this dialog had a strong positive effective, when taken in conjunction with our other initiatives. We could see from the questions we got that at least some Turkers were taking this task very seriously. One Turker, for instance, went to the trouble of collecting all of the Q and A on the whole thread into one consolidated FAQ. He also commented ”Dr. Borthwick is by far the best requester I've ever worked with. Interesting HITs, fair pay + bonuses and good communication. Not sure what else I could ask for.” Anecdotally, we noticed a strong correlation between workers who did a lot

[9] http://turkers.proboards.com/index.cgi

of HITs with high accuracy and workers who were frequent posters on Turker Nation.

Finally, a word on the financial model we used for this project. As can be seen from Table 4, we devoted about 20% of our budget to worker bonuses. We also strove to fulfill the philosophy of "equal pay for equal work" by dividing the HITs into batches according to how many attributes workers had to judge (ranging from 2 - 10, although we omit the "10 attribute" row from this table), so we decreased the pay as the number of attributes to score decreased. Finally, we strove to keep pay between $3 and $4 per hour, based on the advice of Amazon salesmen that this was the maximum hourly rate that yielded significant benefit on MTurk.

# attrs. to score	# HITs	pay per HIT	pay for work	bonus	Amazon fee	total cost	effective hourly pay
9	123	0.22	135.30	27.72	16.30	179.32	2.57
8	188	0.21	197.40	77.17	27.45	302.02	3.26
7	264	0.20	264.00	69.30	33.33	366.63	3.19
6	330	0.18	297.00	114.93	41.19	453.12	2.55
5	423	0.17	359.55	84.15	44.37	488.07	3.69
4	563	0.15	422.25	36.45	45.87	504.57	4.86
3	600	0.12	360.00	129.60	48.96	538.56	5.33
2	704	0.07	246.40	59.89	30.62	336.91	3.45
Total	3195		2281.9	599.21	288.111	3169.221	

Table 4. Annotation statistics for the clustering task

5 Evaluation Methodology

For the evaluation of the clustering task we used the B-Cubed metrics [5]. These metrics were introduced in our task in WePS-2 and have been proved to be the only ones, among the different families of clustering metrics, that satisfy the intuitive formal constraints for this problem [2].

B-Cubed metrics independently compute the precision and recall associated to each item in the distribution. The precision of one item represents the amount of items in the same cluster that belong to its category. Analogously, the recall of one item represents how many items from its category appear in its cluster.

In WePS-2 an extended version of B-Cubed [2] was used to handle the problem of evaluating overlapping clustering (a clustering task where an element can belong to more than one cluster, in our case, when document mentions multiple people with the same ambiguous name). Due to the choices made in the design of the WePS-3 testbed, we excluded the possibility of an overlapping clustering (a document can only belong to one of the reference people or to someone else) and hence we used the original version of the metric [10].

The harmonic mean ($Fmeasure_{\alpha=_{0,5}}$) of B-Cubed Precision and B-Cubed Recall was used for the ranking of systems. For each query we have evaluated the clustering of documents mentioning two different people[11].

In the clustering annotation, a document with 3 or more votes (as explained in Section 4, each document was annotated by five Mechanical Turk workers) for person A, person B or "someone else" was considered as a positive document for the corresponding class. The system's output was evaluated and averaged the B-Cubed Precision and Recall values considering each element classified as person A or person B, over the set of elements classified as person A, person B or "someone else". Note that B-Cubed allows us evaluate the system's clustering solutions even though we do not have a full clustering assessment for each person name. The reason for this is that B-Cubed evaluates on the element level, and unlike Purity/Inverse Purity metrics, we do not have to choose a representative class for each cluster in the output.

For the attribute extraction task participating systems were evaluated based on the attributes they attached to the most representative cluster for each of the people annotated in the clustering gold standard. The cluster with the best recall of attributes for a person in the system output was considered its representative[12] For instance, if we are evaluating "person A" of the name "Tiffany Hopkins", first we rank the clusters in the system output by their attributes recall to "person A" and then we evaluate precision and recall of the attributes in the best ranked cluster. The rationale for using attributes recall as selection criterion is the following: a user confronted with the system output is likely to choose the cluster that exposes the more attributes that identify the person.

Since the method used for the attribute extraction evaluation was pooling the system outputs, recall is not guaranteed to be representative on the attribute annotations: there might be attribute values which are not detected by any system.

6 Participations and evaluation results

The WePS-3 organization was contacted by 34 teams expressing their interest in the clustering task. Out of these, 8 teams submitted a total of 27 different runs. Two baseline systems were included in the evaluation: "all-in-one" which places all documents in a single cluster, and "one-in-one" which places each document in a separate cluster.

[10] Note that results for B-Cubed extended and regular B-Cubed are identical on a non-overlapping clustering.

[11] With the exception of 50 names from the computer science conference names, for which only documents about one person where considered

[12] We also considered the cluster F-measure as an criterion for choosing the representative cluster. We found that this method often missed the cluster with more relevant attributes, resulting in extremely low evaluation results.

Many systems (YHBJ, AXIS, TALP, WOLVES) [12, 7, 9, 11] include Hierarchical Agglomerative Clustering (HAC) as part of their system pipeline. DAEDALUS [14] intentionally departs from the usage of HAC and experiments with the k-Medioids clustering method. In TALP [9] three clustering methods (Lingo, HAC, and 2-steps HAC) where compared using basic features extracted from the web pages.

WOLVES [11] trained a pairwise model to predict the likelihood that two documents refer to the same person. A variety of document features were used (words, named entities, Wikipedia topics, person attributes) along with different pairwise features that measure the similarity between documents (cosine, overlap, Jaccard index, etc). Then a clustering algorithm used these predictions to group the documents. The clustering methods used include HAC and Markov Clustering.

YHBJ [12] concentrates on the document representation and feature weighting. It uses Wikipedia entries to extend a feature set based on bag-of-words and named entities. The assignment of weights to the different features goes beyond the widely used TFIDF metrics, considering the relevance of the features to the name query and how representative it is of the main text of the page.

AXIS [7] analyzed patterns of Web graph structure as part of a two-stage clustering algorithm that also incorporates content-based features. The detection of related web pages is used to overcome the lack of information about Web graph structure.

RGAI [13] represented every document as vector of extracted person attribute values and proceed to apply a clustering algorithm (their experiments include bottom-up clustering and the Xmeans algorithm).

Table 5 presents the results of the 8 participants and the 2 baseline clustering systems. B-Cubed Precision, Recall and F-measure values are macro-averaged over each person name[13]. In the cases where a team submitted multiple run we have chosen the run with the best score as the team representative in the ranking. The table of results shows that:

- The best scoring system obtains balanced results in both precision and recall, while the rest of the participants have biased scores towards one or other metric. Note that the macro-averaged F-measured scores are lower compared to the F-measure that would be obtained using directly the macro-averaged Precision and Recall values. This indicates that, even though Precion or Recall may obtain a high average value, it is usually at the cost of a low score in the other metric. The Unanimous Improvement Ratio results[14] confirmed that only the top two systems in the ranking make a robust improvements (independent of the weighting of Precision and Recall). According to UIR

[13] Note that these tables reflect the scores obtained taking into account only two people for each person name. For this reason this table should not be directly compared to previous WePS campaigns.

[14] The Unanimous Improvement Ratio (UIR) checks, for each system pair, to what extent the improvement is robust across potential metric weighting schemes (see [1]). This measure was also employed in WEPS2 campaign [4].

YHBJ_2 makes a robust improvement of RGAI_AE_1, BYU and TALP_5; and AXIS_2 show a robust improvement compared to BYU.
- As in the previous WePS campaigns, the correct selection of a cluster stopping criterion is a key factor in the performance of systems. The unbalance of Precion and Recall highlighted in the previous point shows how this affects the performance of the clustering systems in WePS.
- Unlike previous WePS campaigns almost all the systems obtained scores above the baselines. It is likely that the *one-in-one baseline* is obtaining lower scores given that we are only considering two people for each name and that these two people are generally well represented in the dataset. This procedure excludes many people with only one document in the Web, which usually rewards the *one-in-one* approach.

Table 6 presents the results for the Attribute Extraction task. In this task Intelius (http://www.intelius.com) provided a baseline system that was evaluated along with the participants. Both RGAI [13] and WOLVES [11] relied on a rule base approaches, tailoring different heuristics for each attribute type.

		Macro-averaged Scores		
		F-measure	B-Cubed	
rank	run	$\alpha =_{,5}$	Pre.	Rec.
1	YHBJ_2_unofficial	0.55	0.61	0.60
2	AXIS_2	0.50	0.69	0.46
3	TALP_5	0.44	0.40	0.66
4	RGAI_AE_1	0.40	0.38	0.61
5	WOLVES_1	0.40	0.31	0.80
6	DAEDALUS_3	0.39	0.29	0.84
7	BYU	0.38	0.52	0.39
	one_in_one_baseline	0.35	1.00	0.23
8	HITSGS	0.35	0.26	0.81
	all_in_one_baseline	0.32	0.22	1.00

Table 5. Clustering results: official team ranking

7 Conclusions

The WePS-3 campaign has continued the research effort on people search by offering a larger testbed, integrating the clustering and attribute extraction task and including the participation of experts from companies. The evaluation has featured the use of Mechanical Turk to achieve a large amount of annotated data, and in this process we have learnt about the oportunities and dangers of such powerful tool. Participant teams in the campaign have further expanded the variety of approaches to the people search problem by including external

	Macro-averaged Scores		
	F-measure		
run	$\alpha =_{,5}$	Pre.	Rec.
RGAI_AE_3	0.18	0.22	0.24
RGAI_AE_1	0.15	0.18	0.19
Intelius_AE_unofficial	0.13	0.16	0.17
RGAI_AE_2	0.12	0.16	0.15
RGAI_AE_4	0.12	0.15	0.16
RGAI_AE_5	0.12	0.15	0.15
BYU	0.10	0.11	0.14
WOLVES_AE_1	0.10	0.18	0.09
WOLVES_AE_2	0.06	0.08	0.07

Table 6. Attribute Extraction results: official team ranking

sources of knowledge (Wikipedia), applying new clustering methods to the task and new feature weighting schemes.

8 Acknowledgments

This work has been partially supported by the Spanish Ministry of Science and Innovation within the project QEAVis-Catiex (TIN2007-67581-C02-01).

We would like to thank Intelius for sharing their expertise in the field of people search and for their collaboration on the task design and the gold standard creation. The economic support made possible such a large scale annotation effort.

References

1. E. Amigó, J. Gonzalo, and J. Artiles. Combining evaluation metrics via the unanimous improvement ratio and its application in weps clustering task. In *2nd Web People Search Evaluation Workshop (WePS 2009), 18th WWW Conference*, 2009.
2. E. Amigó, J. Gonzalo, J. Artiles, and F. Verdejo. A comparison of extrinsic clustering evaluation metrics based on formal constraints. *Information Retrieval*, 2008.
3. J. Artiles, J. Gonzalo, and S. Sekine. The semeval-2007 weps evaluation: Establishing a benchmark for the web people search task. In *Proceedings of the Fourth International Workshop on Semantic Evaluations (SemEval-2007)*. ACL, 2007.
4. J. Artiles, J. Gonzalo, and S. Sekine. Weps 2 evaluation campaign: overview of the web people search clustering task. In *2nd Web People Search Evaluation Workshop (WePS 2009), 18th WWW Conference*, 2009.
5. A. Bagga and B. Baldwin. Entity-based cross-document coreferencing using the vector space model. In *Proceedings of the 17th international conference on Computational linguistics*. ACL, 1998.
6. C. Callison-Burch. Fast, cheap, and creative: evaluating translation quality using Amazon's Mechanical Turk. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 1-Volume 1*, pages 286–295. Association for Computational Linguistics, 2009.

7. K. A. Elena Smirnova and B. Trousse. Using web graph structure for person name disambiguation. In *Third Web People Search Evaluation Forum (WePS-3), CLEF 2010*, 2010.
8. J. G. D. S. Enrique Amig, Javier Artiles and L. Bing. Weps-3 evaluation campaign: Overview of the on-line reputation management task. In *Third Web People Search Evaluation Forum (WePS-3), CLEF 2010*, 2010.
9. D. Ferrs and H. Rodrguez. Talp at weps-3 2010. In *Third Web People Search Evaluation Forum (WePS-3), CLEF 2010*, 2010.
10. J. Hoskins. personal communication from Amazon sales representative, 2010.
11. C. O. Iustin Dornescu and T. Lesnikova. Cross-document coreference for weps. In *Third Web People Search Evaluation Forum (WePS-3), CLEF 2010*, 2010.
12. C. Long and L. Shi. Web person name disambiguation by relevance weighting of extended feature sets. In *Third Web People Search Evaluation Forum (WePS-3), CLEF 2010*, 2010.
13. I. T. Nagy and R. Farkas. Person attribute extraction from the textual parts of web pages. In *Third Web People Search Evaluation Forum (WePS-3), CLEF 2010*, 2010.
14. J. V.-R. Sara Lana-Serrano and J.-C. Gonzlez-Cristbal. Daedalus at webps-3 2010: k-medoids clustering using a cost function minimization. In *Third Web People Search Evaluation Forum (WePS-3), CLEF 2010*, 2010.
15. S. Sekine and J. Artiles. Weps2 attribute extraction task. In *2nd Web People Search Evaluation Workshop (WePS 2009), 18th WWW Conference*, 2009.
16. R. Snow, B. O'Connor, D. Jurafsky, and A. Ng. Cheap and fast—but is it good?: evaluating non-expert annotations for natural language tasks. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 254–263. Association for Computational Linguistics, 2008.